

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-282656

(43) 公開日 平成11年(1999)10月15日

(51) Int.Cl. <sup>8</sup>	識別記号	F I
G 0 6 F 9/06	4 1 0	G 0 6 F 9/06 4 1 0 Q
13/00	3 5 1	13/00 3 5 1 H
13/10	3 2 0	13/10 3 2 0 A

審査請求 未請求 請求項の数 8 O L (全 13 頁)

(21) 出願番号 特願平10-83856

(22) 出願日 平成10年(1998) 3 月30日

(71) 出願人 000005267

ブラザー工業株式会社

愛知県名古屋市長穂区苗代町15番1号

(72) 発明者 今井 正明

愛知県名古屋市長穂区苗代町15番1号

ブラザー工業株式会社内

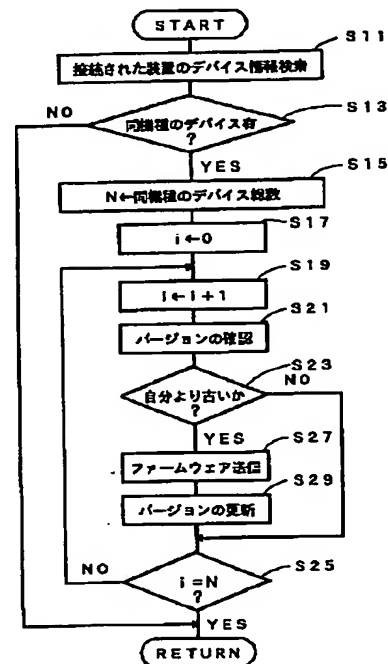
(74) 代理人 弁理士 足立 勉

(54) 【発明の名称】 プログラム書換装置、ネットワークシステム、及び記憶媒体

(57) 【要約】

【課題】 ネットワークに接続された装置のプログラムの書き換えを容易に実行可能とするプログラム書換装置、ネットワークシステム、及び記憶媒体の提供。

【解決手段】 ネットワークに接続された自身と同機種の全てのデバイスを検索し (S11~S15)、各デバイス (i=1~N) に対して、そのデバイスのファームウェアが自身のファームウェアより古いかな否かを判断する (S23)。そして、古い場合には (S23: YES)、自身のファームウェアによってそのデバイスのファームウェアを書き換える (S27)。このため、ネットワークに接続されたデバイスのファームウェアの書き換えを容易に実行することができ、自身と同機種の全てのデバイスに対して、そのファームウェアを少なくとも自身が記憶したファームウェアと同じかそれよりも新しいものとすることができる。従って、ネットワークシステムの管理に要する労力が大幅に低減できる。



## 【特許請求の範囲】

【請求項1】 ネットワークを介してデータの送受信を行う通信手段と、

ソフトウェアプログラムを記憶した記憶手段と、  
を備え、上記ネットワークに接続された他の装置に記憶されたソフトウェアプログラムと、上記記憶手段に記憶されたソフトウェアプログラムとの間で、プログラムの書き換えを行うプログラム書換装置であって、

上記他の装置に対して、上記記憶手段に記憶されたソフトウェアプログラムと同種のソフトウェアプログラムが書き換え可能に記憶されているか否かを判断する種類判断手段と、

該種類判断手段が上記同種のソフトウェアプログラムが書き換え可能に記憶されていると判断したとき、そのソフトウェアプログラムと上記記憶手段に記憶されたソフトウェアプログラムとの新旧を判断する新旧判断手段と、

該新旧判断手段が、上記他の装置に記憶された上記ソフトウェアプログラムの方が上記記憶手段に記憶されたソフトウェアプログラムより古いと判断したとき、上記他の装置に記憶された上記ソフトウェアプログラムを上記記憶手段に記憶されたソフトウェアプログラムによって書き換える第1書換手段と、

を備えたことを特徴とするプログラム書換装置。

【請求項2】 ネットワークを介してデータの送受信を行う通信手段と、

ソフトウェアプログラムを書き換え可能に記憶した記憶手段と、

を備え、上記ネットワークに接続された他の装置に記憶されたソフトウェアプログラムと、上記記憶手段に記憶されたソフトウェアプログラムとの間で、プログラムの書き換えを行うプログラム書換装置であって、

上記他の装置に対して、上記記憶手段に記憶されたソフトウェアプログラムと同種のソフトウェアプログラムが記憶されているか否かを判断する種類判断手段と、

該種類判断手段が上記同種のソフトウェアプログラムが記憶されていると判断したとき、そのソフトウェアプログラムと上記記憶手段に記憶されたソフトウェアプログラムとの新旧を判断する新旧判断手段と、

該新旧判断手段が、上記他の装置に記憶された上記ソフトウェアプログラムの方が上記記憶手段に記憶されたソフトウェアプログラムより新しいと判断したとき、上記記憶手段に記憶された上記ソフトウェアプログラムを上記他の装置に記憶されたソフトウェアプログラムによって書き換える第2書換手段と、

を備えたことを特徴とするプログラム書換装置。

【請求項3】 上記記憶手段が、上記ソフトウェアプログラムを書き換え可能に記憶すると共に、上記新旧判断手段が、上記他の装置に記憶された上記ソフトウェアプログラムの方が上記記憶手段に記憶されたソフトウェア

プログラムより新しいと判断したとき、上記記憶手段に記憶された上記ソフトウェアプログラムを上記他の装置に記憶されたソフトウェアプログラムによって書き換える第2書換手段を、

更に備えたことを特徴とする請求項1記載のプログラム書換装置。

【請求項4】 上記種類判断手段が、上記ネットワークに接続された全ての他の装置に対して上記判断を行い、上記新旧判断手段が、上記種類判断手段が上記同種のソフトウェアプログラムが記憶されていると判断した全ての上記他の装置に対して、上記判断を行うことを特徴とする請求項1～3のいずれかに記載のプログラム書換装置。

【請求項5】 上記記憶手段に記憶されたソフトウェアプログラムに基づき、被記録媒体に画像を形成する画像形成手段を、更に備えたことを特徴とする請求項1～4のいずれかに記載のプログラム書換装置。

【請求項6】 請求項1～5のいずれかに記載のプログラム書換装置と、ソフトウェアプログラムを記憶した他の装置とを、ネットワークを介して接続してなるネットワークシステムであって、

上記プログラム書換装置または上記他の装置の少なくともいずれか一つが、自身が記憶したソフトウェアプログラムの書き換えを禁止する禁止手段を備え、

該禁止手段により書き換えが禁止されているソフトウェアプログラムに対しては、上記第1書換手段または上記第2書換手段が、上記書き換えを行わないことを特徴とするネットワークシステム。

【請求項7】 ネットワークを介してデータの送受信を行う通信手段と、ソフトウェアプログラムを記憶した記憶手段と、を備えた処理装置によって読み取り可能なソフトウェアプログラムを記憶した記憶媒体であって、上記ネットワークに接続された他の装置に対して、上記記憶手段に記憶されたソフトウェアプログラムと同種のソフトウェアプログラムが書き換え可能に記憶されているか否かを判断する種類判断処理と、

該種類判断処理によって上記同種のソフトウェアプログラムが書き換え可能に記憶されていると判断されたとき、そのソフトウェアプログラムと上記記憶手段に記憶されたソフトウェアプログラムとの新旧を判断する新旧判断処理と、

該新旧判断処理によって、上記他の装置に記憶された上記ソフトウェアプログラムの方が上記記憶手段に記憶されたソフトウェアプログラムより古いと判断されたとき、上記他の装置に記憶された上記ソフトウェアプログラムを上記記憶手段に記憶されたソフトウェアプログラムによって書き換える第1書換処理と、

を実行させるソフトウェアプログラムを記憶したことを特徴とする記憶媒体。

【請求項8】 ネットワークを介してデータの送受信を

行う通信手段と、ソフトウェアプログラムを書き換え可能に記憶した記憶手段と、を備えた処理装置によって読み取り可能なソフトウェアプログラムを記憶した記憶媒体であつて、

上記ネットワークに接続された他の装置に対して、上記記憶手段に記憶されたソフトウェアプログラムと同種のソフトウェアプログラムが記憶されているか否かを判断する種類判断処理と、

該種類判断処理によって上記同種のソフトウェアプログラムが記憶されていると判断されたとき、そのソフトウェアプログラムと上記記憶手段に記憶されたソフトウェアプログラムとの新旧を判断する新旧判断処理と、

該新旧判断処理によって、上記他の装置に記憶された上記ソフトウェアプログラムの方が上記記憶手段に記憶されたソフトウェアプログラムより新しいと判断されたとき、上記記憶手段に記憶された上記ソフトウェアプログラムを上記他の装置に記憶されたソフトウェアプログラムによって書き換える第2書換処理と、

を実行させるソフトウェアプログラムを記憶したことを特徴とする記憶媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、ネットワークに接続された他の装置に記憶されたソフトウェアプログラムと自身が記憶したソフトウェアプログラムとの間でプログラムの書き換えを行うプログラム書換装置、そのプログラム書換装置を備えたネットワークシステム、及び、上記プログラム書換装置を実現するための記憶媒体に関する。

【0002】

【従来の技術】パーソナルコンピュータ（以下、パソコンという）やそのパソコンにネットワーク等を介して接続されたプリンタ等の周辺機器は、自身が実行する処理のソフトウェアプログラム（以下、単にプログラムという）をROM等の記憶手段に記憶している。また、近年では、特殊な操作をしたときに書き換え可能となるフラッシュROM等にプログラムを記憶しておき、バグを修正したり新しい機能を追加したりするときにプログラムを書き換えることも考えられている。

【0003】

【発明が解決しようとする課題】ところが、このようなプログラムの書き換えは、管理者が要否を判断してパソコンやプリンタの1台毎に個々に行う必要があり、非常に面倒であつた。例えば、近年では、複数のプリンタと複数のパソコンとをネットワークを介して接続したプリントシステムが考えられている。このようなプリントシステムでは、一つのパソコンから画像データを送信したとき、他のパソコンによって使用されていないプリンタを用いてその画像データを処理することができる。この場合、ネットワークに接続された全てのプリンタに対し

て同様にプログラムの書き換えを実行しなければならない事態もしばしば発生するが、そのようなときにもプリンタの1台毎に個々にプログラムを書き換えなければならない。このため、プリントシステム等のネットワークシステムの管理には多大な労力を必要としていた。

【0004】そこで、本発明は、ネットワークに接続された装置のプログラムの書き換えを容易に実行することのできるプログラム書換装置、ネットワークシステム、及びそれらを実現するための記憶媒体を提供することを目的としてなされた。

【0005】

【課題を解決するための手段及び発明の効果】上記目的を達するためになされた請求項1記載の発明は、ネットワークを介してデータの送受信を行う通信手段と、ソフトウェアプログラムを記憶した記憶手段と、を備え、上記ネットワークに接続された他の装置に記憶されたソフトウェアプログラムと、上記記憶手段に記憶されたソフトウェアプログラムとの間で、プログラムの書き換えを行うプログラム書換装置であつて、上記他の装置に対して、上記記憶手段に記憶されたソフトウェアプログラムと同種のソフトウェアプログラムが書き換え可能に記憶されているか否かを判断する種類判断手段と、該種類判断手段が上記同種のソフトウェアプログラムが書き換え可能に記憶されていると判断したとき、そのソフトウェアプログラムと上記記憶手段に記憶されたソフトウェアプログラムとの新旧を判断する新旧判断手段と、該新旧判断手段が、上記他の装置に記憶された上記ソフトウェアプログラムの方が上記記憶手段に記憶されたソフトウェアプログラムより古いと判断したとき、上記他の装置に記憶された上記ソフトウェアプログラムを上記記憶手段に記憶されたソフトウェアプログラムによって書き換える第1書換手段と、を備えたことを特徴とする。

【0006】このように構成された本発明のプログラム書換装置は、通信手段を介してネットワークに接続され、他の装置とデータの送受信を行うことができる。また、本発明は、記憶手段にプログラムを記憶している。更に、種類判断手段は、上記他の装置に対して、上記記憶手段に記憶されたプログラムと同種のプログラムが書き換え可能に記憶されているか否かを判断し、同種のプログラムが書き換え可能に記憶されていると判断されたとき、新旧判断手段が、そのプログラムと上記記憶手段に記憶されたプログラムとの新旧を判断する。そして、新旧判断手段が、上記他の装置に記憶されたプログラムの方が上記記憶手段に記憶されたプログラムより古いと判断したとき、第1書換手段は、上記他の装置に記憶されたプログラムを上記記憶手段に記憶されたプログラムによって書き換える。

【0007】このため、本発明をネットワークに接続すれば、記憶手段に記憶されたプログラムと同種のプログラムを書き換え可能に記憶した他の装置に対して、プロ

プログラムの新旧を判断し、記憶手段に記憶されたプログラムより古ければ、その記憶手段に記憶されたプログラムによって他の装置のプログラムを自動的に書き換えることができる。従って、本発明をネットワークに接続すれば、ネットワークに接続された装置のプログラムの書き換えを容易に実行することができ、そのネットワークシステムの管理に要する労力を大幅に低減することができる。

【0008】請求項2記載の発明は、ネットワークを介してデータの送受信を行う通信手段と、ソフトウェアプログラムを書き換え可能に記憶した記憶手段と、を備え、上記ネットワークに接続された他の装置に記憶されたソフトウェアプログラムと、上記記憶手段に記憶されたソフトウェアプログラムとの間で、プログラムの書き換えを行うプログラム書換装置であって、上記他の装置に対して、上記記憶手段に記憶されたソフトウェアプログラムと同種のソフトウェアプログラムが記憶されているか否かを判断する種類判断手段と、該種類判断手段が上記同種のソフトウェアプログラムが記憶されていると判断したとき、そのソフトウェアプログラムと上記記憶手段に記憶されたソフトウェアプログラムとの新旧を判断する新旧判断手段と、該新旧判断手段が、上記他の装置に記憶された上記ソフトウェアプログラムの方が上記記憶手段に記憶されたソフトウェアプログラムより新しいと判断したとき、上記記憶手段に記憶された上記ソフトウェアプログラムを上記他の装置に記憶されたソフトウェアプログラムによって書き換える第2書換手段と、を備えたことを特徴とする。

【0009】このように構成された本発明のプログラム書換装置は、通信手段を介してネットワークに接続され、他の装置とデータの送受信を行うことができる。また、本発明は、記憶手段にプログラムを書き換え可能に記憶している。更に、種類判断手段は、上記他の装置に対して、上記記憶手段に記憶されたプログラムと同種のプログラムが記憶されているか否かを判断し、同種のプログラムが記憶されていると判断されたとき、新旧判断手段が、そのプログラムと上記記憶手段に記憶されたプログラムとの新旧を判断する。そして、新旧判断手段が、上記他の装置に記憶されたプログラムの方が上記記憶手段に記憶されたプログラムより新しいと判断したとき、第2書換手段は、上記記憶手段に記憶されたプログラムを上記他の装置に記憶されたプログラムによって書き換える。

【0010】このため、本発明をネットワークに接続しておけば、そのネットワークに、上記記憶手段に記憶されたものと同種のプログラムを記憶した他の装置が新たに接続され（他の装置のプログラムが書き換えられた場合でもよい）、しかも、そのプログラムが記憶手段に記憶されたプログラムより新しい場合、上記他の装置に記憶されたプログラムによって記憶手段のプログラムを自

動的に書き換えることができる。従って、本発明によってネットワークシステムを構成すれば、ネットワークに接続された装置のプログラムの書き換えを容易に実行することができ、そのネットワークシステムの管理に要する労力を大幅に低減することができる。

【0011】請求項3記載の発明は、請求項1記載の構成に加え、上記記憶手段が、上記ソフトウェアプログラムを書き換え可能に記憶すると共に、上記新旧判断手段が、上記他の装置に記憶された上記ソフトウェアプログラムの方が上記記憶手段に記憶されたソフトウェアプログラムより新しいと判断したとき、上記記憶手段に記憶された上記ソフトウェアプログラムを上記他の装置に記憶されたソフトウェアプログラムによって書き換える第2書換手段を、更に備えたことを特徴とする。

【0012】すなわち、本発明は、請求項1記載の構成と請求項2記載の構成とを併せ持っている。このため、本発明をネットワークに接続したときも、本発明を用いて構成されたネットワークシステムに新たな装置が接続されたときも（本発明または他の装置のプログラムが書き換えられた場合でもよい）、上記同種のプログラムをより新しい方に自動的に書き換えることができる。従って、請求項1記載の発明の効果、及び請求項2記載の発明の効果に加えて、その相乗効果によりネットワークシステムの管理を一層容易にして、その管理に要する労力を一層低減することができるといった効果が生じる。

【0013】請求項4記載の発明は、請求項1～3のいずれかに記載の構成に加え、上記種類判断手段が、上記ネットワークに接続された全ての他の装置に対して上記判断を行い、上記新旧判断手段が、上記種類判断手段が上記同種のソフトウェアプログラムが記憶されていると判断した全ての上記他の装置に対して、上記判断を行うことを特徴とする。

【0014】本発明では、種類判断手段がネットワークに接続された全ての他の装置に対して上記判断を行い、新旧判断手段が、種類判断手段が同種のプログラムが記憶されていると判断した全ての上記他の装置に対して上記判断を行う。すると、新旧判断手段の判断結果に基づき、前述のようにプログラムの書き換えが行われる。このため、本発明では、請求項1～3のいずれかに記載の発明の効果に加えて、上記プログラムの書き換えをネットワークに接続された全ての装置に対して自動的に実行することができるといった効果が生じる。

【0015】例えば、本発明の構成を備えた請求項1記載の発明では、ネットワークに接続された全ての装置の上記同種のプログラムを、少なくとも記憶手段に記憶されたプログラムと同じかそれよりも新しいものとする。また、本発明の構成を備えた請求項2記載の発明では、記憶手段に記憶されたプログラムを、ネットワークに接続された全ての装置の上記同種のプログラムの内、最も新しいものとする。更に、本

発明の構成を備えた請求項3記載の発明では、ネットワークに接続された本発明を含む全ての装置の上記同種のプログラムを、各装置の上記同種のプログラムの内、最も新しいものとする事ができる。

【0016】また、本発明の構成を備えた請求項3記載の発明において、第2書換手段による書き換えがなされたときにはネットワークに接続された全ての上記他の装置に対して同様の処理をやり直すようにすれば、第2書換手段によって更新された記憶手段のプログラムによって、引き続き、上記他の装置のプログラムを更新し直すことができる。この場合、ネットワークに接続された本発明を含む全ての装置の上記同種のプログラムを、各装置の上記同種のプログラムの内、最も新しいものとする事が一層確実に実行できる。

【0017】なお、上記全ての他の装置とは、ネットワークがブリッジ、ルータ、ゲートウェイ等によって複数のセグメント（管理エリアとも呼ばれる）に分割されている場合、本発明と同一のセグメントに接続された全ての他の装置を指している。請求項5記載の発明は、請求項1～4のいずれかに記載の構成に加え、上記記憶手段に記憶されたソフトウェアプログラムに基づき、被記録媒体に画像を形成する画像形成手段を、更に備えたことを特徴とする。

【0018】本発明では、記憶手段に記憶されたプログラムに基づき、画像形成手段によって被記録媒体に画像を形成することができる。すなわち、本発明は、請求項1～4のいずれかに記載の発明を画像形成装置に適用したものである。この種の画像形成装置では、画像形成手段の機械的な動作に要する時間がプログラムの処理に要する時間に比べてきわめて長い。このため、前述のプリントシステムのように、ネットワークに複数の装置を接続しておき、使用されていない装置によって画像を形成可能にする要望がきわめて高い。この場合、ネットワークに接続された全ての装置に対して同様のプログラムの書き換えを実行しなければならない事態も比較的頻繁に発生する。

【0019】本発明では、このような画像形成装置に対して請求項1～4のいずれかに記載の発明を適用しているので、ネットワークに接続された装置のプログラムの書き換えを容易にするといった各発明の効果が一層顕著に現れる。請求項6記載の発明は、請求項1～5のいずれかに記載のプログラム書換装置と、ソフトウェアプログラムを記憶した他の装置とを、ネットワークを介して接続してなるネットワークシステムであって、上記プログラム書換装置または上記他の装置の少なくともいずれかが、自身が記憶したソフトウェアプログラムの書き換えを禁止する禁止手段を備え、該禁止手段により書き換えが禁止されているソフトウェアプログラムに対しては、上記第1書換手段または上記第2書換手段が、上記書き換えを行わないことを特徴とする。

【0020】本発明では、請求項1～5のいずれかに記載のプログラム書換装置または上記他の装置の少なくともいずれかが禁止手段を備え、その禁止手段により書き換えが禁止されているプログラムに対しては、上記第1書換手段または上記第2書換手段が上記書き換えを行わない。前述のように、請求項1～5のいずれかに記載のプログラム書換装置とプログラムを記憶した他の装置とをネットワークを介して接続した場合、上記同種のプログラムはより新しいものへと書き換えられるが、本発明では、禁止手段を備えた装置におけるそのような書き換えを禁止することができる。このため、古いプログラムでなければ処理できないデータ等がある場合、一部の装置でそのプログラムを書き換えずに保持することができる。

【0021】請求項7記載の発明は、ネットワークを介してデータの送受信を行う通信手段と、ソフトウェアプログラムを記憶した記憶手段と、を備えた処理装置によって読み取り可能なソフトウェアプログラムを記憶した記憶媒体であって、上記ネットワークに接続された他の装置に対して、上記記憶手段に記憶されたソフトウェアプログラムと同種のソフトウェアプログラムが書き換え可能に記憶されているか否かを判断する種類判断処理と、該種類判断処理によって上記同種のソフトウェアプログラムが書き換え可能に記憶されていると判断されたとき、そのソフトウェアプログラムと上記記憶手段に記憶されたソフトウェアプログラムとの新旧を判断する新旧判断処理と、該新旧判断処理によって、上記他の装置に記憶された上記ソフトウェアプログラムの方が上記記憶手段に記憶されたソフトウェアプログラムより古いと判断されたとき、上記他の装置に記憶された上記ソフトウェアプログラムを上記記憶手段に記憶されたソフトウェアプログラムによって書き換える第1書換処理と、を実行させるソフトウェアプログラムを記憶したことを特徴とする。

【0022】このため、本発明に記憶されたプログラムを、ネットワークを介してデータの送受信を行う通信手段とプログラムを記憶した記憶手段とを備えた処理装置に読み取らせれば、請求項1記載の種類判断手段、新旧判断手段、及び第1書換手段に相当する種類判断処理、新旧判断処理、及び第1書換処理を実行させることができる。従って、請求項1記載のプログラム書換装置を一般のコンピュータ等の処理装置によって容易に実現することができ、請求項1記載の発明と同様の効果が生じる。

【0023】請求項8記載の発明は、ネットワークを介してデータの送受信を行う通信手段と、ソフトウェアプログラムを書き換え可能に記憶した記憶手段と、を備えた処理装置によって読み取り可能なソフトウェアプログラムを記憶した記憶媒体であって、上記ネットワークに接続された他の装置に対して、上記記憶手段に記憶され

たソフトウェアプログラムと同種のソフトウェアプログラムが記憶されているか否かを判断する種類判断処理と、該種類判断処理によって上記同種のソフトウェアプログラムが記憶されていると判断されたとき、そのソフトウェアプログラムと上記記憶手段に記憶されたソフトウェアプログラムとの新旧を判断する新旧判断処理と、該新旧判断処理によって、上記他の装置に記憶された上記ソフトウェアプログラムの方が上記記憶手段に記憶されたソフトウェアプログラムより新しいと判断されたとき、上記記憶手段に記憶された上記ソフトウェアプログラムを上記他の装置に記憶されたソフトウェアプログラムによって書き換える第2書換処理と、を実行させるソフトウェアプログラムを記憶したことを特徴とする。

【0024】このため、本発明に記憶されたプログラムを、ネットワークを介してデータの送受信を行う通信手段とプログラムを書き換え可能に記憶した記憶手段とを備えた処理装置に読み取らせれば、請求項2記載の種類判断手段、新旧判断手段、及び第2書換手段に相当する種類判断処理、新旧判断処理、及び第2書換処理を実行させることができる。従って、請求項2記載のプログラム書換装置を一般のコンピュータ等の処理装置によって容易に実現することができ、請求項2記載の発明と同様の効果が生じる。

#### 【0025】

【発明の実施の形態】次に、本発明に好適な実施の形態について、図面に基いて説明する。なお、以下に説明する実施の形態は、端末装置としてのプリンタを複数台含むと共に、それぞれのプリンタを用いて印刷処理を行うパソコンを複数含んだネットワークシステムに対して本発明を適用した場合の実施の形態である。

【0026】先ず、本実施の形態のネットワークシステムSの構成について、図1を用いて説明する。図1に示すように、ネットワークシステムSは、電話回線またはLAN等のネットワークWと、そのネットワークWを介してデータの送受信を行う通信手段としての複数のNIC1と、端末装置としてのプリンタ10、30、40と、ネットワークWに接続されたパソコン50、60と、プリンタ10に接続された複数のコンピュータ20と、プリンタ40に並列に接続されたイメージスキャナ70と、プリンタ40に直列に接続された用紙ソータ80とにより構成されている。ここで、プリンタ10、30及び40は異なる機種種のプリンタである。なお、パソコン50は、CPU51、ROM52、RAM53を備えた周知の構成を有し、パソコン60も同様に構成されている。

【0027】また、NIC1は、返信手段としてのトランシーバ2と、LANコントローラ3と、共有メモリ4と、CPU5と、ROM6と、RAM7と、フラッシュROM(FlashROM)8と、NVRAM9とをバスBで接続して構成されている。ここで、RAM7は、

その内部に端末情報エリア7aを備えている。

【0028】一方、プリンタ10は、CPU11と、ROM12と、RAM13と、フラッシュROM14と、NVRAM15と、出力インタフェース(出力I/F)17と、入力インタフェース(入力I/F)18とをバスBで接続して構成されている。また、出力インタフェース17には画像形成手段としての印字部19が接続されている。ここで、RAM13は、その内部に出力バッファ13aと入力バッファ13bとを備えている。また、プリンタ10は、バスBに接続された接続ラインJを介してNIC1に接続されると共に、入力インタフェース18を介して各コンピュータ20に接続されている。

【0029】次に、ネットワークシステムSにおける概要動作について図1を用いて説明する。なお、以下の概要動作の説明では、NIC1とプリンタ10との間における処理について説明するが、他のNIC1とプリンタ30、40等との間においても同様の処理が実行される。

【0030】パソコン50内のCPU51は、プリンタ10またはそれに接続されているNIC1に送信すべきデータを生成し、ネットワークWを介してNIC1のトランシーバ2に送信する。このとき、当該データの送信は、NIC1にて処理されるべきデータはTFTP(トリリアルファイル転送プロトコル)で、プリンタ10にて処理されるべきデータはLPR(ライン・プリンタ・リモート)で、処理装置を特に指定する必要がないデータはUDP(ユーザーデータグラム転送プロトコル)等の適宜のプロトコルで、それぞれ実行される。そして、データを受信したトランシーバ2は、これを復調し、LANコントローラ3を介してバスBに出力する。ここで、当該LANコントローラ3は、ネットワークWを介したパソコン50との間におけるデータの送受信を制御する。

【0031】次に、CPU5は、NIC1が受信したデータがTFTPで送信されてきたときは、当該データをROM6及びフラッシュROM8に記憶されている情報を用いて処理する。このとき、RAM7は、当該CPU5における処理に必要な情報を読み出し可能に一時的に記憶する。その後、CPU5は、処理した結果をバスB、LANコントローラ3、トランシーバ2及びネットワークWを介して送信元(パソコン50、60または他のNIC1)に返信する。

【0032】一方、NIC1が受信したデータがLPRで送信されてきたときは、CPU5は、当該データを共有メモリ4及び接続ラインJを介してプリンタ10に転送する。その際、共有メモリ4にデータを書き込んだ後、CPU5は、図示しない信号線を通してCPU11に対してインターラプト(割り込み指令)を発生させ、当該データの処理を実行させる。なお、上述したデータ

10

20

30

40

50



に対するCPU5の処理に必要な制御プログラムは、ROM6に予め記憶されている。

【0033】ここで、共有メモリ4は、情報処理の際にNIC1とプリンタ10との間で共有すべき情報を一時的に記憶しておくためのメモリである。次に、NIC1から転送したプリンタ10において処理すべきデータ

(例えば、パソコン50または60からLPRで送信されてきたデータ)が接続ラインJを介してプリンタ10に入力されると、CPU11は、当該データをバスBを介して取得した後、ROM12及びフラッシュROM14に記憶されている情報を用いて処理する。その後、CPU11は、処理した結果をバスB、接続ラインJ、及び共有メモリ4を介してNIC1に返信する。このとき、RAM13は当該CPU11における処理に必要な情報を読み出し可能に一時的に記憶する。このデータに対するCPU11の処理に必要な制御プログラムは、ROM12またはフラッシュROM14に予め記憶されている。

【0034】更に、プリンタ10において処理すべきデータに対する処理結果をプリンタ10から受領したNIC1は、その処理結果をそのままネットワークWを介してパソコン50、60または他のプリンタ30、40に転送する。一方、各コンピュータ20から出力されたプリンタ10において印字出力すべき画像等のデータは、入力インタフェース18を介してプリンタ10内に取り込まれ、バスBを介して入力バッファ13b内に記憶され、印刷データに展開処理されて出力バッファ13a内に一時的に記憶された後、再びバスBを介して出力インタフェース17から印字部19に出力される。すると、印字部19は、当該データに対応した画像を被記録媒体としての用紙等に形成する。更に、パソコン50または60から出力された印字出力すべき画像等のデータをプリンタ10へ入力したときは、当該データがネットワークW経由でNIC1を介してプリンタ10の入力バッファ13bに取り込まれた後、上記と同様に処理される。このプリンタ10本来の印字出力処理は、ROM12及びフラッシュROM14に記憶されている制御プログラムに基づくCPU11の制御の下で実行される。

【0035】なお、ネットワークシステムS内に含まれている各プリンタ10、30、及び40は共通のNIC1を備えているが、プリンタ自体としては異なる機種のものを含んでいる。具体的には、印刷機構や取り扱いデータの相違等に対応して、それぞれのプリンタのROM12には後述のデバイス情報が記憶(後述のように、バージョン等はNVRAM15に記憶)されているが、そのデバイス情報は各プリンタ毎に異なったものとなっている。また、ネットワークWには、NIC1とは異なる機種種のNICを介して図示しない他のプリンタも接続されている。

【0036】次に、フラッシュROM8、14は周知の

ように書き換え可能に構成されている。そこで、この部分に記憶されたファームウェア等の制御プログラムを書き換えることにより、プリンタ10の機能をバージョンアップすることができる。続いて、このバージョンアップに関わる処理について説明する。

【0037】図2は、プリンタ10がネットワークWに接続されたとき、そのCPU11が実行する処理を表すフローチャートである。処理を開始すると、CPU11は、先ず、S11(Sはステップを表す：以下同様)にて、ネットワークWに接続された全ての装置(プリンタ30~40、イメージスキャナ70等：以下デバイスと総称する)のデバイス情報を検索する。なお、ここで全てのデバイスとは、ネットワークWがブリッジ、ルータ、ゲートウェイ等によって複数のセグメントに分割されている場合、プリンタ10と同一のセグメントに接続された全ての他のデバイスを指している。従って、例えばインターネット等を介して接続された世界中のデバイスを指すわけではない。

【0038】図3は、上記S11の処理の詳細を表すフローチャートである。図3に示すように、S11では、先ず、S111にて、デバイス情報要求のためのブロードキャストパケットを、ネットワークWを介して接続された全てのデバイスに送信する。続くS113では、応答監視期間として、少なくとも3秒間計時可能なタイマをスタートする。更に、続くS115では、ネットワークWを介して各プリンタからデバイス情報を受信し、受信したデバイス情報をRAM13内の所定領域にリストアップしていく。そして、このS115の処理をタイマをスタートしてから3秒経過するまで繰り返し(S117)、3秒経過すると(S117:YES)、続くS13(図2)へ移行する。

【0039】ここで、プリンタ30等の各種デバイスは、上記ブロードキャストパケットを受信すると乱数を発生し、その乱数に対応した数msec、待機した後、デバイス情報を返信する。このため、プリンタ10には、上記乱数に応じたバラバラのタイミングで、各デバイスからのデバイス情報がネットワークWを介して返信される。なお、ここでいうデバイス情報とは、後述のデバイス情報100(図4)のような詳細な情報ではなく、ID等のようにデバイスの機種が特定できる情報であればよい。このため、各種デバイスは、印刷等の処理中にもその処理を停止することなく上記デバイス情報を返信することができる。

【0040】図2へ戻って、S13では、S115でリストアップしたデバイス情報(ID)に基づき、プリンタ10と同機種のデバイスがあるか否かを判断する。同機種のデバイスがない場合は(S13:NO)、そのまま処理を終了し、同機種のデバイスがある場合は(S13:YES)、S15以下の処理へ移行する。

【0041】ここでは、同機種のデバイスの総数をNに

代入し(S15)、変数*i*を0にリセットし(S17)、続いて、*i*を一つインクリメントする(S19)。以上の処理により、最初は*i*=1となる。続くS21では、S115でリストアップした内の*i*番目のデバイスに対し、バージョンの確認を行う。この処理では、上記*i*番目のデバイスに対して詳細なデバイス情報を要求し、この要求に対して、各種デバイスは図4に例示するようなデバイス情報100を返信する。

【0042】図4に示すように、デバイス情報100は、レーザプリンタ、カラープリンタ等のデバイスの種別を表すデバイスクラス101、デバイスの機種毎に付与された型番等のデバイス名102、プリンタ10の状態を表すステータス103、及びファームウェアのバージョンを表すプリンタバージョン104からなるプリンタ10側のデバイス情報と、NIC1側のデバイス情報としてのNICデバイス名105及びNICバージョン106と、そのプリンタ10に付与されたIPアドレス107とを羅列して構成されている。また、デバイスクラス101、デバイス名102、ステータス103、プリンタバージョン104、NICデバイス名105、及びNICバージョン106のデータの先頭には、各々のデータの長さを表すデバイスクラスレングス111、デバイス名レングス112、ステータスレングス113、プリンタバージョンレングス114、NICデバイス名レングス115、及びNICバージョンレングス116が添付されている。

【0043】S21では、受信したデバイス情報100の内、特にプリンタバージョン104等に基づいて、フラッシュROM14に記憶されたファームウェアのバージョンを確認する。図2に戻って、続くS23では、そのバージョンを比較して、上記デバイスのファームウェアが自身のフラッシュROM14が記憶しているファームウェアより古いかなかを判断する。そして、ファームウェアが自身のものより新しい場合、または同じ場合は(S23:NO)、S25へ移行する。S25では、*i*=Nとなったかなかを判断し、*i*≠Nの場合は(S25:NO)、前述のS19へ移行する。そして、*i*をインクリメントした後、その*i*番目のデバイスに対してS21～S25の処理を実行する。

【0044】一方、S23にて、*i*番目のデバイスのファームウェアが自分よりも古い(YES)と判断したとき、S27、S29へ順次移行して、次のような処理を実行する。まず、S27では、自身のフラッシュROM14に記憶されたファームウェアを上記*i*番目のデバイスへ送信する。このファームウェアのデータには、ヘッダとして、フラッシュROM14に書き換えるべきファームウェアであることを示すデータが添付されており、これを受信したデバイスは、ROM12に予め記憶された書換プログラムに基づいて、フラッシュROM14のファームウェアを受信したファームウェアに書き換え

る。なお、上記ファームウェアのデータを受信したとき、その受信側のデバイスが印刷等の処理中であることも考えられるが、この場合は印刷等の処理が一旦終了してからファームウェアを書き換えるのが望ましい。この間、上記ファームウェアのデータは、プリンタ10側で保持してもデバイス側で保持してもよい。こうすることによって、印刷中に、例えば1ページ目と2ページ目とでフォントが切り替わる等の事態が回避できる。

【0045】また、書き換えの対象となるデバイス側から、書換要求が送信され、プリンタ10側でこの要求を判別した時点で初めて、そのデバイスに対して上記ファームウェアのデータを送信するようにしてもよい。これにより、この書換要求が送られて来ないデバイスに関しては、上記ファームウェアのデータを送らないようにしてもよいわけである。

【0046】続くS29では、上記デバイスにおけるファームウェアの書き換えが成功したことを確認した後、上記デバイスのバージョンを更新する。すなわち、上記デバイスは、ファームウェアの書き換えが終了すると、そのファームウェアのチェックSUMをプリンタ10へ返信する。CPU11は、そのチェックSUMが正しいことを確認した上で、デバイス側のNVRAM15に記憶されたバージョン情報(デバイス情報100のプリンタバージョン104に対応)を、新たに書き換えられたファームウェアに対応したものに更新するように応答する。S29の次は前述のS25へ移行し、同機種のデバイスで未処理のものがある場合(S25:NO)、S19へ移行して同様の処理を繰り返す。そして、同機種の全デバイスに対してS19～S25の処理が完了すると、S25にて肯定判断して処理を終了する。

【0047】このように、本実施の形態のプリンタ10では、ネットワークWに接続された自身と同機種の全てのデバイスに対して、そのフラッシュROM14に記憶されたファームウェアが自身のフラッシュROM14に記憶されたファームウェアより古い場合には、自身のファームウェアによってそのデバイスのファームウェアを自動的に書き換えることができる。このため、ネットワークWに接続されたデバイスのファームウェアの書き換えを容易に実行することができ、自身と同機種の全てのデバイスに対して、そのファームウェアを少なくとも自身が記憶したファームウェアと同じかそれよりも新しいものとすることができる。従って、ネットワークシステムSの管理に要する労力を大幅に低減することができる。特に、プリンタ10、30、40等の画像形成装置では、印字部19の機械的な動作に要する時間がプログラムの処理に要する時間に比べてきわめて長い。このため、ネットワークWに多数のプリンタを接続しておき、使用されていないプリンタによって画像を形成可能にする要望がきわめて高い。この場合、ネットワークWに接続された全てのプリンタに対して同様のプログラムの書



き換えを実行しなければならぬ事態も比較的頻繁に発生する。プリンタ10では、前述のように、ネットワークWに接続された全ての同機種のプリンタに対して、必要に応じて自動的にファームウェアの書き換えを行うことができるので、ネットワークWに接続されたデバイスのファームウェアの書き換えを容易にするといった上記効果が一層顕著に現れる。

【0048】なお、上記実施の形態では、S11（詳しくはS111）にてデバイス情報としてIDのみを要求しているが、図4に例示したような詳細なデバイス情報100を要求してもよい。この場合、S21にてデバイス情報100を要求する必要がなくなる。但し、上記実施の形態ではネットワークWに接続された各種デバイスに対して自身と同機種か否かを判断し、同機種のデバイスに対してのみデバイス情報100を要求しているので、通信に要する時間を短縮することができる。また、上記実施の形態では、自身と同機種であれば（S13：YES）同種のファームウェアがフラッシュROM14に書き換え可能に記憶されていることを前提としている。同機種であっても一部のプリンタでファームウェアの形態が全く異なったり、そのファームウェアを書き換え不能に記憶していたりする可能性がある場合は、同機種のデバイスか否かの判断（S13）に続いて同種のファームウェアが書き換え可能に記憶されているか否かを改めて判断してもよい。

【0049】更に、上記実施の形態では、ネットワークWに接続された他のデバイスのファームウェアが自身のファームウェアより古い場合に（S23：YES）そのデバイスのファームウェアを書き換えているが（S27）、他のデバイスのファームウェアが自身のファームウェアより新しい場合には自身のファームウェアを書き換えるようにしてもよい。図5はそのような実施の形態におけるCPU11の処理を表すフローチャートである。なお、図5では、図2と同様の処理については同様の符号を付した。そこで、図2の処理との相違点についてのみ説明する。

【0050】S23に代わるS33では、i番目のデバイスのファームウェアバージョンと自身のファームウェアバージョンとの新旧を比較する。そして、同じである場合は前述のS25へ移行し、i+1番目のデバイスに対して同様の処理を行う。S33にてデバイスのファームウェアが自身のファームウェアよりも古いと判断したときは、前述のS27、S29にてそのデバイスのファームウェアを書き換えた後、前述のS25へ移行する。更に、S33にてデバイスのファームウェアが自身のファームウェアよりも新しいと判断したときは、次のS35へ移行する。

【0051】S35では、そのデバイスにファームウェアを送信するよう要求を送信する。続くS37では、上記要求に対して返信されたファームウェアによって、自

身のフラッシュROM14に記憶されたファームウェアを書き換える。なお、この書き換えに関する書換プログラムも、ROM12に予め記憶されている。そして、前述のように書き換えが成功したことを判断すると、S39にて自身のNVRAM15に記憶されたバージョン情報を更新して、前述のS25へ移行する。

【0052】本実施の形態では、自身のフラッシュROM14に記憶されるファームウェアを、ネットワークWに接続された全てのプリンタのファームウェアの内、最も新しいものとする事ができる。なお、一般的な傾向として、後からネットワークWに接続されるデバイスほど新しいファームウェアを記憶している可能性が高い。そこで、図5の処理は、プリンタ10の電源投入時、または、所定時間毎（例えば5～6時間毎）に実行すると一層効果的である。すると、電源投入毎または所定時間毎に自身のファームウェアをネットワークシステムSにおける最新のものとすることができる（S37）。この場合、各デバイス（i=1～Nに対応）に対してS19～S25の処理を順次実行する途中で自身のファームウェアが更新されたとしても（S37）、次の処理タイミングでは、他の全てのプリンタのファームウェアを同様の最新のものとすることができる（S27）。

【0053】更に、図6に示す処理のように、自身のファームウェアを更新したときは（S37）、前述のS39に続いてS17へ移行するようにしてもよい。この場合、自身のファームウェアを更新した後（S37）、iを0にリセットして全てのデバイスに対してファームウェアの更新処理（S19～S25）がやり直される。このため、図6の処理では、ネットワークWに接続されたプリンタ10を含む全デバイスのファームウェアを最新のものとすることが、一回の処理によってきわめて容易に、かつ迅速に実行できる。

【0054】また、ネットワークWに接続された全てのデバイス（少なくともプリンタ10と同機種の全てのデバイス）が電源投入毎または所定時間毎に図2の処理を実行するようにしてもよく、デバイスのネットワークWへの接続時にそのデバイスが各デバイスに自身のIDを送信するようにして、そのIDを受信したときにもプリンタ10が図2の処理を実行するようにしてもよい。これらの場合にも、最終的には、プリンタ10と同機種の全デバイスのファームウェアを最新のものとすることができる。更に、これらのタイミングで全てのデバイスが処理を実行する場合、図5の処理においてS27、S29を省略してもよい。すなわち、i番目のデバイスのファームウェアが自身のファームウェアより新しい場合にのみS35～S39の処理を実行し、他の場合は直接S25へ移行してもよい。この場合にも、全てのデバイスのファームウェアを最新のものとすることができる。

【0055】なお、上記各実施の形態では、S27またはS37の処理に対応するファームウェアの書き換えに

失敗した場合の説明を省略したが、この場合の処理としては、例えば次の2通りの形態が考えられる。一つはそのまま処理を終了してリセットする形態で、もう一つは、成功するまで何度も書き換えを繰り返す形態である。また、書き換えを所定回数繰り返しても成功しなかった場合に、書き換えに失敗した旨を表示してリセットする形態も考えられる。これらの形態は、プリンタ10の設定によって適宜選択できるようにしてもよい。

【0056】また、プリンタ10等が受信するデータには、古いファームウェアでなければ処理できないデータが含まれる可能性もある。例えば、使われなくなった外字に関するフォントデータ等は新しいファームウェアに含まれない場合もある。そこで、S27またはS35の直前に次のような処理を挿入することにより、一部のプリンタでそのプログラムを書き換えずに保持することも可能である。

【0057】図7は、S27またはS35の直前に挿入すべき上記処理の一例を表すフローチャートである。この処理を挿入した場合、S23またはS33にてファームウェアの新旧を比較し、ファームウェアの書き換えが必要と判断すると（例えばS23: YES）、S41にて書き換えが禁止されているか否かを判断する。すなわち、S27またはS37の処理によってフラッシュROM14が書き換えられる装置において、その書き換えを禁止する設定がなされているか否かを判断する。禁止されていない場合は（S41: NO）、そのまま前述のS27またはS35へ移行し、禁止されている場合は（S41: YES）、S43へ移行する。

【0058】S43では、その新旧のファームウェアの相違点が、バグの修正のみであるか否かを判断する。バグの修正のみである場合は（S43: YES）、ファームウェアを書き換えてもなんら不都合は生じないので、S27またはS35へ移行してファームウェアを書き換える。また、バグの修正以外にもファームウェアが変更されている場合は、ファームウェアを書き換えることなくS25へ移行する。

【0059】このような処理を挿入することによって、上記書き換えの禁止が設定された一部のデバイスで、ファームウェアを実質的に書き換えずに保持することができる。また、バグの修正は実行されるので、一層エラーの少ない処理を実現することができる。なお、S43における判断は、ファームウェア同士を直接比較して行ってもよく、ROM12等にデータテーブル等を保持してそれを参照してもよく、インターネット等を介してソフトウェア提供会社に問い合わせてもよい。

【0060】更に、S43の処理は省略して、書き換えが禁止されていれば（S41: YES）、直接S25へ移行するようにしてもよい。この場合、上記禁止が設定されたデバイスでは、ファームウェアを一切書き換えることなく保持することができる。また、上記禁止に関わ

る処理は、ファームウェアを書き換える書換プログラムの側で実行してもよい。

【0061】なお、上記各実施の形態において、S13が種類判断処理に、S23及びS33が新旧判断処理に、S27が第1書換処理に、S37が第2書換処理に、S41が禁止処理に、それぞれ相当し、それらの処理を記憶したROM12の記憶領域及びCPU11が、種類判断手段、新旧判断手段、第1書換手段、第2書換手段、及び禁止手段に、それぞれ相当する。また、プリンタ10及びそれに接続されたNIC1が本発明のプログラム書換装置に、ROM12が本発明の記憶媒体に、それぞれ相当し、フラッシュROM14が記憶手段に相当する。

【0062】更に、本発明は上記実施の形態になんら限定されるものではなく、本発明の要旨を逸脱しない範囲で種々の形態で実施することができる。例えば、S11ではネットワークWに接続された全てのデバイスに対してデバイス情報を要求しているが（S111）、図示しない操作パネル等によって指定されたり、パソコン50等から入力されたりすることによって得られたデバイスのリストに指示された一部のデバイスに対してのみデバイス情報を要求してもよい。また、S111では、自分と同機種で、かつ自分よりも新しいタイプのデバイスである場合にのみデバイス情報を返信するように要求してもよい。更に、S117ではデバイス情報の受信（S115）を3秒で打ち切っているが、全てのデバイスからデバイス情報が返信されるまで受信を継続してもよい。

【0063】また、プリンタ10では、ROM12に記憶された書換プログラムによってフラッシュROM14を書き換えているが、書換プログラムはフラッシュROM14に記憶しておいてもよい。この場合、書換プログラムをRAM13またはNVRAM15にコピーした後、そのコピーされたプログラムによって上記書き換えが実行される。更に、この場合、ROM12を省略して一層のコストダウンを図ることができる。

【0064】また更に、上記各処理は、NIC1のフラッシュROM8に記憶されたファームウェアの書き換えに対しても、同様に適用することができる。更に、本発明は、プリンタ10等の画像形成装置を中心としたネットワークシステム（いわゆるプリントシステム）に限らず、通信カラオケシステム等、他の種々のネットワークシステムに適用することができる。また記憶媒体としては、ROM、RAM等の素子の他、種々の形態が考えられる。例えば、CD-ROM、フロッピーディスク等でもよく、カードスロットへ挿入可能なプログラムカートリッジ等でもよく、インターネット上のファイルサーバであってもよい。更に、本発明のプログラム書換装置は、ファームウェア等のソフトウェアプログラムをプリンタ、パソコン等の装置との間で送受信するファイルサーバやパソコンであってもよい。

## 【図面の簡単な説明】

【図1】本発明が適用されたネットワークシステムの構成を表すブロック図である。

【図2】そのシステムのプリンタで実行される処理を表すフローチャートである。

【図3】その処理の一部を詳細に表すフローチャートである。

【図4】上記処理で使用するデバイス情報のデータ構成を表すブロック図である。

【図5】上記処理の変形例を表すフローチャートである。

【図6】上記処理の他の変形例を表すフローチャートである。

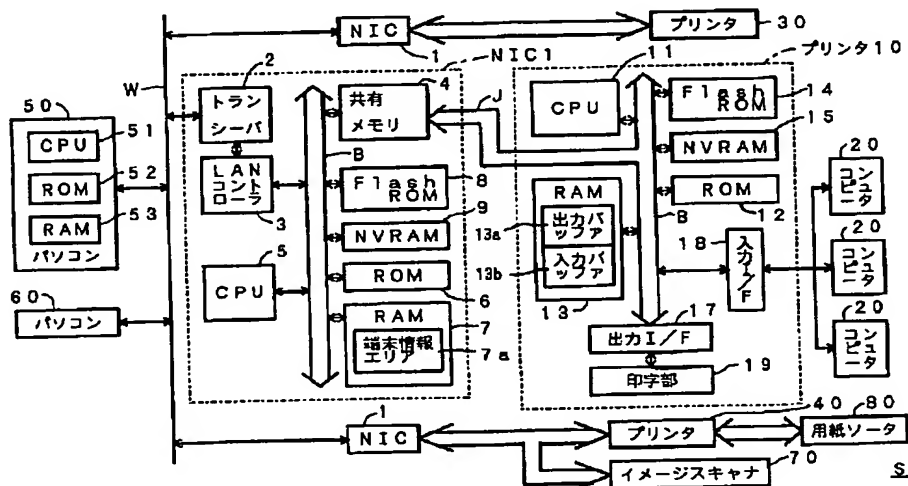
\* ある。

【図7】上記処理の更に他の変形例を表すフローチャートである。

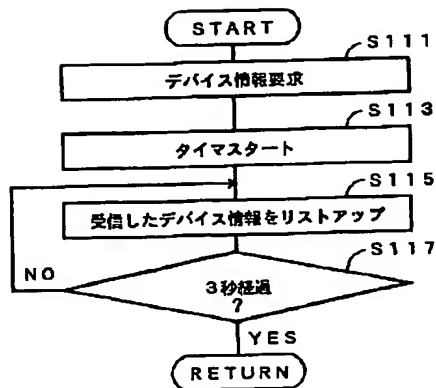
## 【符号の説明】

1…NIC 5, 11…CPU 6,  
12…ROM  
7, 13…RAM 8, 14…フラッシュROM  
9, 15…NVRAM  
10, 30, 40…プリンタ 19…印字部  
50, 60…パソコン  
100…デバイス情報 S…ネットワークシステム  
W…ネットワーク

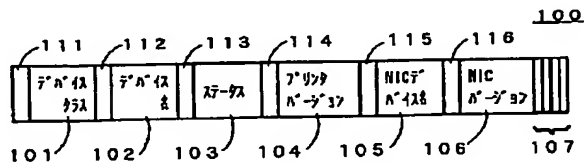
【図1】



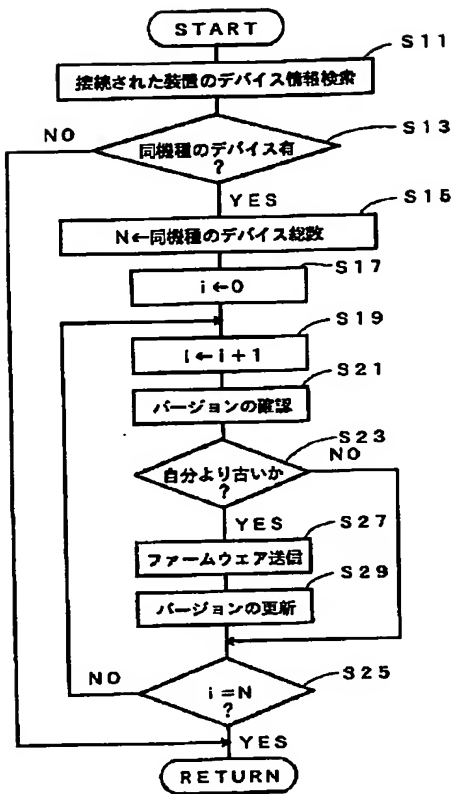
【図3】



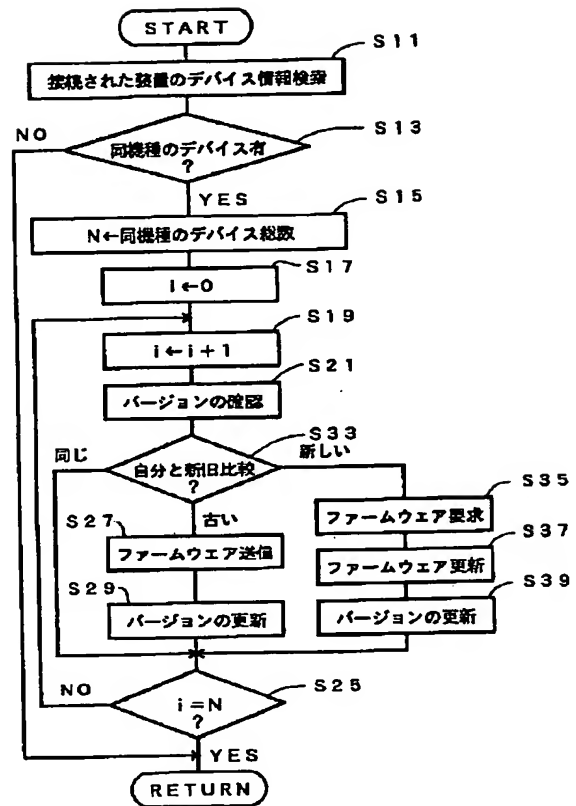
【図4】



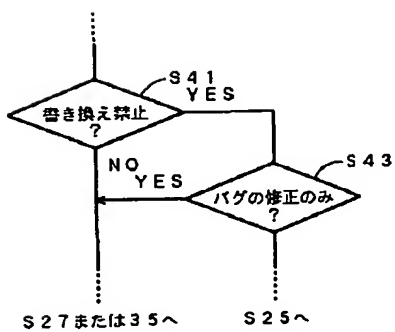
【図2】



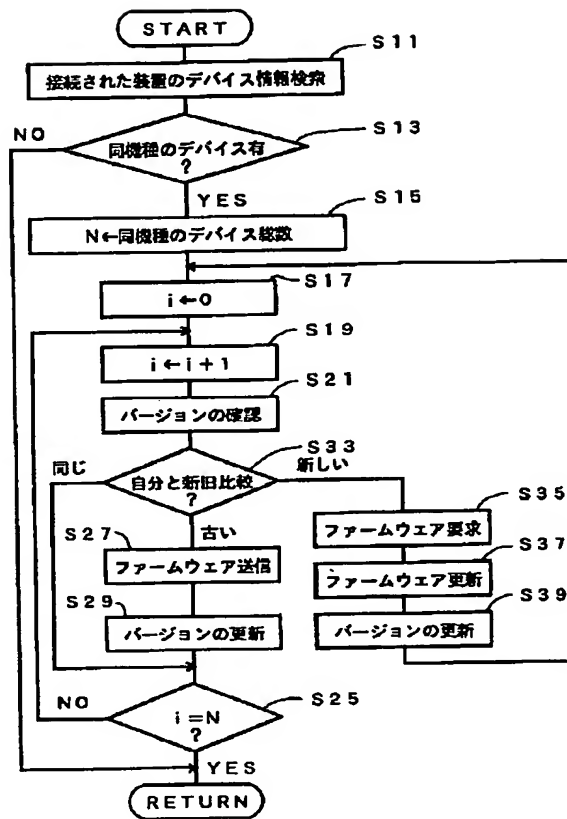
【図5】



【図7】



【図6】



Partial Translation of JP 1999-282656

Publication Date: October 15, 1999

Application No.: 1998-83856

Filing Date: March 30, 1998

Applicant: Brother Industries, Ltd.

15-1, Naeshiro-cho, Mizuho-ku, Nagoya

Inventor: Masaaki IMAI

[0037]

Fig. 2 is a flow chart showing the processing performed by the CPU 11 when the printer 10 is connected to the network W. When the CPU 11 starts the processing, firstly the CPU 11 retrieves the device information of all the devices (printers 30 to 40, an image scanner 70 or the like: hereinafter, generically called a device) connected to the network W at a step S11 (S represents step: hereinafter, similarly). Herein, all the devices mean all the other devices connected to a segment which is same as a segment to which the printer 10 belongs when the network W is divided into a plurality of segments by a bridge, a router and a gateway or the like. Therefore, for example, all the devices do not mean the devices connected via the Internet or the like in the world.

[0038]

Fig. 3 is a flow chart showing the detail of the processing of the above step S11. As shown in Fig. 3, at a step S11, first, broadcasting



packets for requiring the device information are transmitted to all the devices connected via the network W at a step S111. At the following step S113, a timer capable of being clocked is started for at least 3 seconds as a response monitoring period. Then, at the following step S115, the device information is received from each printer via the network W, and the received device information is listed up in a predetermined region in a RAM13. This processing of the step S115 is repeated until three seconds elapse after starting the timer (S117), and when three seconds elapse (S117:YES), the processing is moved to the following step S13 (Fig. 2).

[0039]

Herein, various devices such as the printer 30 generate random numbers when the various devices receive the above broadcasting packet, and reply the device information after waiting for several msec corresponding to the random numbers. Thereby, the device information from each device is replied to the printer 10 via the network W at the scattering timing according to the above random numbers. Herein, the device information is not detailed information such as the device information 100 (Fig. 4) to be described below, but should be information by which the model of the device can be specified such as ID. Thereby, the various devices can reply the above device information without stopping the processing during processing of printing or the like.

[0040]

Returning to Fig. 2, at the step S13, based on the device information (ID) listed up at the step S115, it is judged whether a device of the same model as the printer 10 exists. When the device of the same model does not

exist (S13: NO), the processing is completed as it is. When the device of the same model exists (S13: YES), the processing is moved to the processing of the following step S15.

[0041]

Herein, the total of the device of the same model is substituted for N (S15), and a variable i is reset to 0 (S17). Then the i is incremented (S19). By the above processing, firstly i is equal to 1. At the following step S21, the version of the i-th device listed up at the step S115 is confirmed. In this processing, the detailed device information is required to the above i-th device, and as illustrated in Fig. 4, various devices replies the device information 100 to this requisition.

[0042]

As shown in Fig. 4, the device information 100 is composed by listing device information of the side of the printer 10 containing a device class 101 representing the classification of devices such as a laser printer and a color printer, a device name 102 such as a model number given for every model of the device, a status 103 representing the state of the printer 10, and a printer version 104 representing the version of firmware; an NIC device name 105 and a NIC version 106 as device information of the side of an NIC1; and an IP addresses 107 given to the printer 10. A device class length 111, device name length 112, status length 113, printer version length 114, NIC device name length 115 and NIC version length 116 representing the length of each data, are attached to the head of the data of the device class 101, device name 102, status 103, printer version 104, NIC device name 105 and NIC version 106.

[0043]

At the step S21, the version of the firmware stored in a flash ROM 14 is particularly confirmed among the received device information 100 based on the printer version 104 or the like. Returning to Fig. 2, at the following step S23, the version is compared, and it is judged whether the firmware of the above device is older than that stored in the own flash ROM 14. When the firmware is newer than the own one or is the same (the step S23: NO), the processing is moved to the step S25. At the step S25, it is judged whether i is equal to N and, when i is not equal to N (S25: NO), the processing is moved to the above step S19. After the i is incremented, the processing of the steps S21 to S25 is performed to the i-th device.

[0044]

On the other hand, at the step S23, when it is judged that the firmware of the i-th device is older than the own one (YES), the processing is sequentially moved to the steps S27, S29, and the following processing is performed. Firstly, at the S27, the firmware stored in the own flash ROM 14 is transmitted to the above i-th device. Data showing the firmware which should be rewritten to the flash ROM 14 as a header is attached to the data of this firmware. The device receiving the data rewrites to the firmware receiving the firmware of the flash ROM 14 based on rewriting program stored in a ROM12 previously. When the data of the above firmware is received, it is considered that the device of the reception side is performing processing such as printing. However, in this case, after the processing of printing or the like is completed once, it is preferable to rewrite the firmware. In the meantime, the data of the above firmware

may be held at the side of the printer 10, or may be held at the side of the device. Therefore, for example, situations such as a font change at the 1st page and the 2nd page can be avoided during printing.

[0045]

When a rewriting requisition is transmitted from the side of the device as the object of rewriting and this requisition is distinguished at the side of the printer 10 for the first time, the data of the above firmware may be transmitted to the device. Thereby, the data of the above firmware may not be transmitted for the device to which the rewriting requisition is not transmitted.

[0046]

At the following step S29, after confirming that the rewriting of the firmware in the above device is succeeded, the version of the above device is updated. That is, the above device replies the check SUM of the firmware to the printer 10 after the rewriting of the firmware is completed. The CPU 11 replies so that the version information (corresponds to the printer version 104 of the device information 100) stored in an NVRAM 15 of the side of the device is updated to one corresponding to the firmware newly rewritten after confirming that the check SUM is right. The processing is moved to the above step S25 after the step S29. When an unsettled device of the same model exists (S25: NO), the processing is moved to the step S19, and the same processing is repeated. When the processing of the steps S19 to S25 is completed to all the devices of the same model, affirmative judgment is performed at the step S25, and the processing is completed.

[0047]

Thus, referring to the printer 10 of this embodiment, when the firmware stored in the flash ROM 14 to all the devices of the same model as the own device connected to the network W is older than the firmware stored in the own flash ROM 14, the firmware of the device can be automatically rewritten by the own firmware. For this reason, the rewriting of the firmware of the device connected to the network W can be easily performed. At least the firmware is set to the same one as the firmware stored in the own device, or newer one to all the devices of the same model as the own device. Therefore, labor required for the control of a network system S can be drastically reduced. Particularly, in an image forming device such as the printers 10, 30, 40, the time required for the mechanical operation of the printing part 19 is very long as compared with that required for the processing of the program. Thereby, a request that a lot of printers are connected to the network W and the image can be formed by the printer which is not used is very high. In this case, a situation where the rewriting of the same program must be performed to all the printers connected to the network W is also generated comparatively and frequently. As described above, since the firmware can be automatically rewritten to all the printers of the same model connected to the network W if needed in the printer 10, the above effect that the rewriting of the firmware of the device connected to the network W can be easily performed is more notably shown.

Fig. 2

1. Retrieve device information of continued device
2. Is device of same model?
3.  $N \leftarrow$  Number of device of same model
4. Confirm version
5. Is older than oneself?
6. Transmit firmware
7. Update version

Fig. 3

1. Require device information
2. Start timer
3. List up received device information
4. Elapse 3 seconds



## 【図面の簡単な説明】

【図1】本発明が適用されたネットワークシステムの構成を表すブロック図である。

【図2】そのシステムのプリンタで実行される処理を表すフローチャートである。

【図3】その処理の一部を詳細に表すフローチャートである。

【図4】上記処理で使用されるデバイス情報のデータ構成を表すブロック図である。

【図5】上記処理の変形例を表すフローチャートである。

【図6】上記処理の他の変形例を表すフローチャートで\*

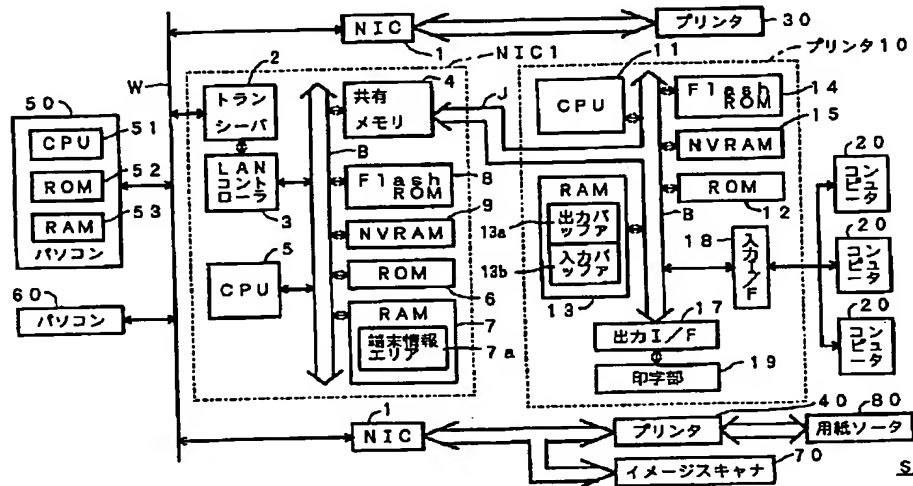
\*ある。

【図7】上記処理の更に他の変形例を表すフローチャートである。

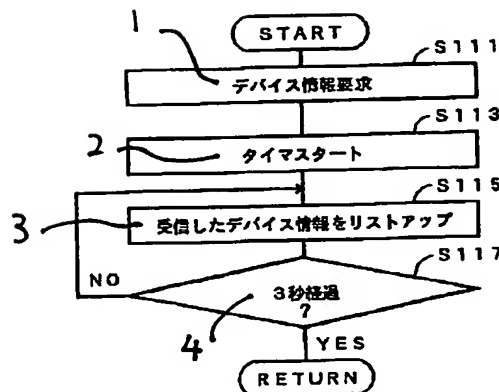
## 【符号の説明】

1…NIC 5, 11…CPU 6,  
12…ROM  
7, 13…RAM 8, 14…フラッシュROM  
9, 15…NVRAM  
10, 30, 40…プリンタ 19…印字部  
50, 60…パソコン  
100…デバイス情報 S…ネットワークシステム  
W…ネットワーク

【図1】



【図3】 Fig. 3



【図4】

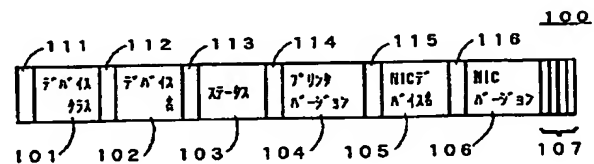
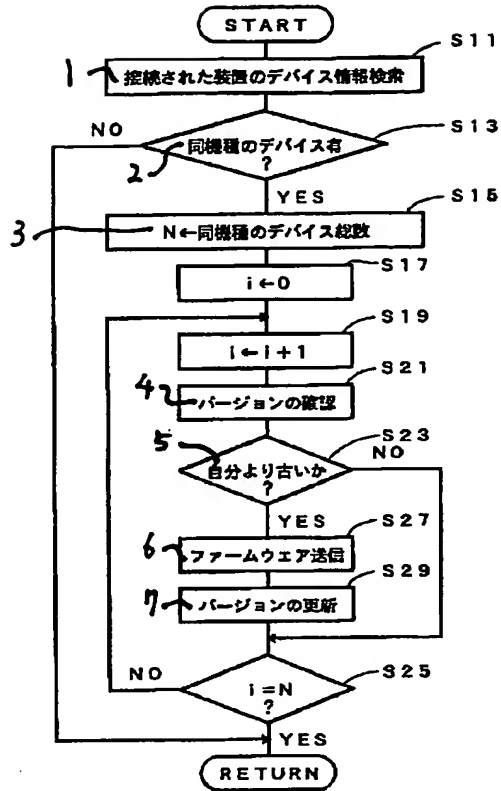
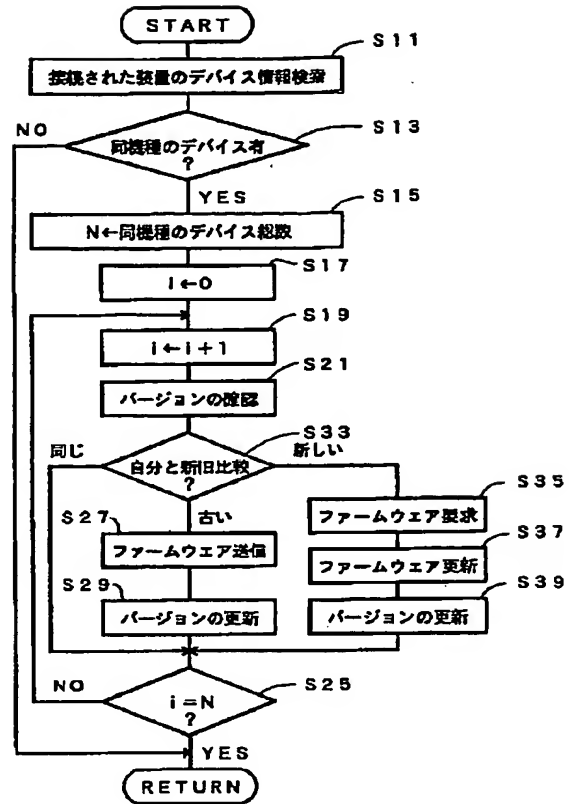


Fig. 2  
【図2】



【図5】



【図7】

